SMFANet: A Lightweight Self-Modulation Feature Aggregation Network for Efficient Image Super-Resolution

Mingjun Zheng*, Long Sun*, Jiangxin Dong, and Jinshan Pan†

School of Computer Science and Engineering,
Nanjing University of Science and Technology, Nanjing, China
{mingjunzheng, cs.longsun, jxdong, jspan}@njust.edu.cn

Abstract. Transformer-based restoration methods achieve significant performance as the self-attention (SA) of the Transformer can explore non-local information for better high-resolution image reconstruction. However, the key dot-product SA requires substantial computational resources, which limits its application in low-power devices. Moreover, the low-pass nature of the SA mechanism limits its capacity for capturing local details, consequently leading to smooth reconstruction results. To address these issues, we propose a self-modulation feature aggregation (SMFA) module to collaboratively exploit both local and non-local feature interactions for a more accurate reconstruction. Specifically, the SMFA module employs an efficient approximation of self-attention (EASA) branch to model non-local information and uses a local detail estimation (LDE) branch to capture local details. Additionally, we further introduce a partial convolution-based feed-forward network (PCFN) to refine the representative features derived from the SMFA. Extensive experiments show that the proposed SMFANet family achieve a better trade-off between reconstruction performance and computational efficiency on public benchmark datasets. In particular, compared to the $\times 4$ SwinIR-light, SMFANet+ achieves **0.14dB** higher performance over five public testsets on average, and $\times$**10** times faster runtime, with only about **43%** of the model complexity (*e.g.,* FLOPs). Our source codes and pre-trained models are available at: https://github.com/Zheng-MJ/SMFANet.

Keywords: Efficient image SR · Self-attention · Feature aggregation

1 Introduction

Single-image super-resolution (SISR) refers to restoring a high-resolution (HR) image from the given degraded low-resolution (LR) counterpart. With the rapid rise of streaming platforms and ultra-high-definition (UHD) content becoming the industrial standard, SISR has attracted significant attention from academia to industry for its ability to provide a pleasurable watching experience at a lower cost of media transmission. Concurrently, the ill-posed nature of SISR

* Equal contribution
† Corresponding author.

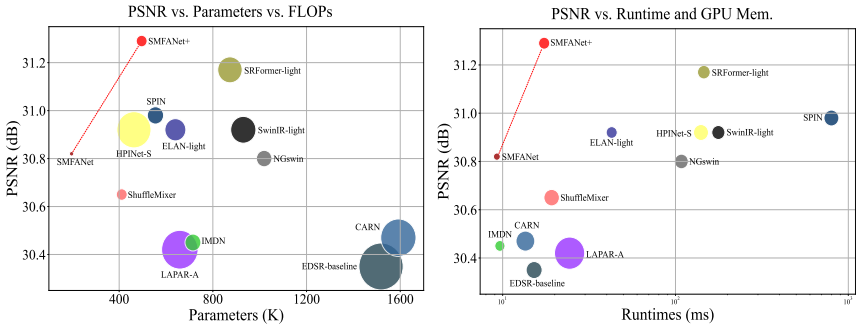


Fig. 1: Overall performance comparison between our proposed method and other state-of-the-art lightweight methods on Manga109 [33] for $\times 4$ SR. The circle sizes in the left and right figures represent the number of FLOPs of the model and the GPU usage required for inference, respectively. The proposed SMFANet family achieve a better trade-off between computational efficiency and reconstruction performance.

task makes it challenging for conventional hand-crafted prior-based [8, 39, 44], or interpolation-based approaches to decently solve this problem.

Over the past decade, deep learning has revolutionized the field of SISR. Various convolutional neural networks (CNNs) [1, 9, 10, 17, 24, 26, 35–37, 50] have been developed to solve this problem. As the basic operation of CNNs, the convolution operator is translation-invariant and has limited receptive fields, which limits its ability to model non-local information. For better image restoration, CNN-based methods [26, 50, 51] become deeper and larger to enhance the representation ability. These high-capacity networks consume high computational costs, *e.g.*, RCAN [50] has 15.59M parameters with over 400 layers, which cannot be applied to source-limited devices, *e.g.*, smart phones.

Recently, the vision Transformer (ViT) [12] achieves impressive success on both high-level [29] and low-level vision tasks [4, 7, 21, 25, 27, 41, 47, 49, 53], as the self-attention (SA) mechanism in the ViT can model non-local information effectively. However, the SA mechanism requires high computational resources with massive memory consumption. To reduce the computational cost, window-based self-attention [7, 25, 29, 49], transposed SA [45], and weight multiplexing strategy [48] have been developed to ease the computational burden. However, these still require a long time to learn feature dependencies for image super-resolution.

Furthermore, recent studies [11, 34] reveal that ViTs favor learning low-frequency components, which leads to smoothed reconstruction results. Thus, these limitations motivate us to think: *Is it possible to develop an efficient feature modulation block that explores non-local information in a SA-like manner while modeling local details for efficient image super-resolution?*

To this end, we propose a self-modulation feature aggregation (SMFA) module that contains an efficient approximation of self-attention (EASA) branch to exploit non-local information, and a local detail estimation (LDE) branch to

model local features. On the EASA branch, we acquire the low-frequency contents through a downsampling operation, calculate the global variance of the input features to modulate the processed low-frequency features, and then use the modulated features to aggregate the input features adaptively. As the EASA prioritizes non-local structure information exploration, we use the LDE branch with convolutional layers to capture local features in parallel.

In addition, we develop an efficient partial convolution-based feed-forward network (PCFN), which further refines the representative features derived from the SMFA in both spatial and channel dimensions. We formulate the proposed SMFA module and PCFN into an end-to-end trainable network, called SMFANet, to solve SISR. Extensive experiments show that the proposed SMFANet achieves a favorable trade-off between computational efficiency and reconstruction performance (see Figure 1).

The main contributions are summarized as follows:

- We develop an efficient SMFA module to extract representative features, where the EASA branch for exploring non-local information and the LDE branch for capturing local features.
- We present a lightweight PCFN to refine the features derived from the SMFA in spatial and channel dimensions further.
- We conduct quantitative and qualitative evaluations for our proposed SMFANet on public benchmark datasets, and the results demonstrate that our method achieves a favorable trade-off between model complexity and reconstruction performance.

2 Related Work

CNN-based super-resolution. SRCNN [9] first introduces effective end-to-end trainable CNNs to map LR directly to its counterpart HR, with superior performance over previous hand-crafted prior and interpolation methods. Then, FSRCNN [10] and ESPCN [35] adopt a post-upsampling strategy to improve efficiency. VDSR [18] uses global residual learning to solve the training problem of deep networks effectively. Therefore, subsequent CNN-based methods can stack deeper and larger layers to capture more information for better image restoration. EDSR [26] increases the model footprint to 43M with significant performance improvement, and RCAN [50], based on channel attention, builds a model with over 400 layers. However, these large models that require high computational costs make them challenging to deploy in real-world applications.

To alleviate the computational burden, many CNN-based methods aim to strike a balance between reconstruction performance and model complexity. CARN [1] applies group convolution and cascades an efficient residual network. IMDN [17] adopts the information multi-distillation blocks to split, refine, and aggregate the feature mapping step-by-step, which significantly reduces the model parameters. Furthermore, its improvement work RFDN [28] is the winning solution in the NTIRE 2022 Efficient SR Challenge. BSRN [24] uses blueprint convolutions to reduce computational redundancy significantly. Unlike the above

approaches that employ progressive feature refinement techniques, recent studies [32, 36, 52] tend to introduce non-local feature modulation to extract representative features for reconstruction. VapSR [52] uses large kernel convolutions in the attention branch to enlarge the receptive field. SAFMN [36] enlarges the receptive field by introducing different down-sampling rates to obtain multi-scale spatial features, and then aggregates these features to generate the modulation map. MDRN [32] employs the parallel implementation of multiple stridden convolutions and polling operations to capture finer global structural information. Although these methods can model non-local feature interactions at a low computational cost and achieve good performance, performing reconstruction using aggregated non-local features only and ignoring local features leads to artifacts in local details (e.g., corners and edges) of the super-resolved image. To avoid this problem, we propose an efficient SMFA module that collaboratively models local and non-local features for more accurate reconstruction.

ViT-based super-resolution. The vision Transformer (ViT) [12] utilizes the self-attention (SA) mechanism to explore global information for image reconstruction and achieves great success. IPT [4] first introduces a standard vision Transformer to solve the image SR problem. However, the SA mechanism of the ViT consumes huge resources, so various attention variants have been proposed to reduce the computational burden. SwinIR [25] employs window self-attention and outperforms the CNN-based large model [26]. ELAN [49] proposes a group-wise self-attention module and shares the weights to reduce complexity significantly. Moreover, recent studies [11, 34] have pointed out that ViTs have the nature of the low-pass filter, resulting in smooth reconstruction results. Therefore, many ViT-based methods [6, 20, 23] utilize convolutional operators to enhance local details for higher-quality image reconstruction. Although these methods integrate the advantages of both convolutional and transformer structures and utilize both local and non-local information to achieve high-quality image reconstruction results, they still rely on window-based attention variants, and feature interactions across windows require much time to perform. Different from these methods, we develop an efficient approximation of self-attention (EASA) branch to model non-local information with much lower model complexity.

3 Proposed Method

Our goal is to develop a simple yet effective CNN model to collaboratively explore local and non-local feature information for accurate image super-resolution. To this end, we first develop a self-modulation feature aggregation (SMFA) module to effectively exploit representative features, where an efficient approximate self-attention (EASA) module based on feature modulation is used for non-local feature interactions, while an additional local detail estimation (LDE) branch is used for local information exploration. To refine the features generated by the SMFA layer, we further introduce a partial convolution-based feed-forward network (PCFN). We incorporate these components into a unified unit that

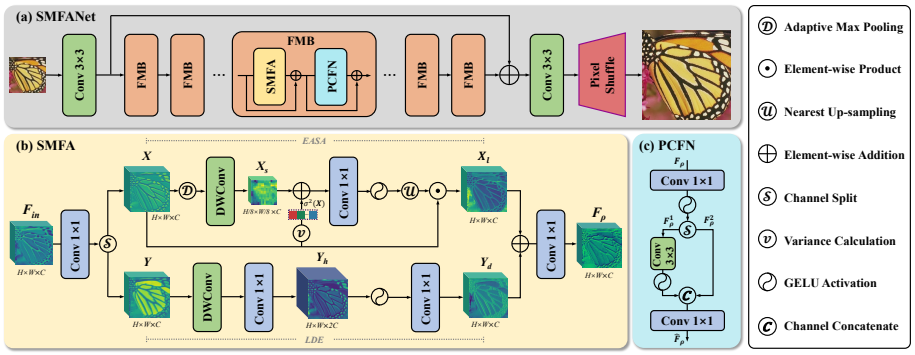


Fig. 2: Network architecture of the proposed SMFANet. The proposed SMFANet consists of a shallow feature extraction module, feature modulation blocks (FMBs), and a lightweight image reconstruction module. Feature modulation block contains one self-modulation feature aggregation (SMFA) module and one partial convolution-based feed-forward network (PCFN).

pays more attention to information-rich regions for better reconstruction. In the following, we explain the main ideas of each component in detail.

3.1 Overall architecture

Figure 2 illustrates the overall architecture of our proposed SMFANet. It takes a low-resolution image as input, and uses a 3×3 convolutional layer to extract shallow features. The extracted shallow features are then fed to a sequence of feature modulation blocks (FMBs) to produce deep representative features, where the FMB consists of a self-modulation feature aggregation (SMFA) module and a partial convolution-based feed-forward network (PCFN). After the feature modulation block, the representative features are processed by an image reconstruction module to reconstruct the high-quality output. To make the reconstruction module as lightweight as possible, a 3×3 convolutional layer is used to convert the channel dimensions to a specific size that adjusts to the upsampling ratio, and a PixelShuffle layer [35] is used for enlargement. To facilitate high-frequency information learning, we insert a global residual connection before the image reconstruction module.

3.2 Self-modulation feature aggregation module

Exploring non-local information is important for image SR reconstruction. Recent ViT-based SR methods [4, 7, 25, 47, 49] utilize various self-attention mechanisms to explore non-local information and achieve impressive reconstruction performance. However, these self-attention variants are computationally expensive and have a limited ability to model local details, as their low-pass filter nature makes them preferentially capture low-frequency information [11, 20, 34].

To address this problem, we develop a lightweight self-modulation feature aggregation (SMFA) module to collaboratively model local and non-local features for accurate reconstruction. Within the SMFA module, an efficient approximation of self-attention (EASA) branch is implemented for non-local information exploration at a moderate cost, and a local detail estimation (LDE) branch is used for capturing local information.

Given the input feature $F_{in} \in \mathbb{R}^{H\times W\times C}$, where $H \times W$ denotes the spatial size and C is the number of channels, we first apply a 1×1 convolution to the normalized F_{in} to expand the channel, and then split the channel into two parts as inputs to the EASA and LDE branches:

$$\{X, Y\} = \mathcal{S}(Conv_{1\times1}(||F_{in}||_2)), \tag{1}$$

where $||\cdot||_2$ is the L_2 normalization, $Conv_{1\times1}(\cdot)$ denotes a 1×1 convolutional layer, $\mathcal{S}(\cdot)$ denotes a channel splitting operation, $X \in \mathbb{R}^{H\times W\times C}$ and $Y \in \mathbb{R}^{H\times W\times C}$. We then process the features X and Y in parallel via the EASA and LED branches, producing the non-local feature X_l and local feature Y_d, respectively. Finally, we fuse X_l and Y_d together with element-wise addition and feed them into a 1×1 convolution to form a representative output of the SMFA module. This process can be formulated as:

$$F_\rho = Conv_{1\times1}(X_l + Y_d), \tag{2}$$

where $F_\rho \in \mathbb{R}^{H\times W\times C}$ is the output feature.

Efficient approximation of self-attention. We obtain the low-frequency components through a downsampling operation and feed them into a 3×3 depth-wise convolution to generate non-local structure information $X_s \in \mathbb{R}^{H/8\times W/8\times C}$:

$$X_s = DWConv_{3\times3}(\mathcal{D}(X)), \tag{3}$$

where $\mathcal{D}(\cdot)$ denotes the adaptive max pooling with a scaling factor of 8, $DWConv_{3\times3}(\cdot)$ is a 3×3 depth-wise convolutional layer. To embed global descriptions for modulating non-local representation X_s, we introduce the variance of X as the statistical divergence of spatial information, and merge it with the non-local representation X_s through a 1×1 convolution:

$$\begin{aligned} \sigma^2(X) &= \frac{1}{N}\sum_{i=0}^{N-1}(x_i - \mu)^2, \\ X_m &= Conv_{1\times1}(X_s + \sigma^2(X)), \end{aligned} \tag{4}$$

where $\sigma^2(X) \in \mathbb{R}^{1\times1\times C}$ is the variance of X, N is the total number of pixels, x_i denotes the value of each pixel, μ is the mean of all pixel values, and $X_m \in \mathbb{R}^{H\times W\times C}$ represents the modulated feature. This variance modulation mechanism facilitates better exploring non-local information, and we study it in detail in Section 5.

Finally, we use the modulated features to aggregate the input feature X for extracting the representative structure information X_l:

$$X_l = X \odot \mathcal{U}(\phi(X_m)), \tag{5}$$

where $\phi(\cdot)$ refers to the GELU activation function [15], $\mathcal{U}(\cdot)$ denotes a nearest upsampling operation, and $\odot$ represents the element-wise product operation.

Local detail estimation. Local details are important for the pleasing high-frequency reconstruction. As the EASA prioritizes non-local structure information exploration, we develop a simple local detail estimation layer to capture local features simultaneously. In detail, an expanded depth-wise convolution with a kernel size of 3×3 is used to encode local information Y_h from the input features Y. Then, we use two 1×1 convolutions with a hidden GELU activation to generate the enhanced local feature Y_d, which is achieved by:

$$\begin{aligned} Y_h &= Conv_{1\times1}(DWConv_{3\times3}(Y)), \\ Y_d &= Conv_{1\times1}(\phi(Y_h)), \end{aligned} \tag{6}$$

where $Y_h \in \mathbb{R}^{H\times W\times 2C}$ is the encoded local information.

3.3 Partial convolution-based feed-forward network

The regular feed-forward network (FFN) [12, 40] operates on each pixel location identically, which lacks the exchange of information in the spatial dimension. Inspired by [5, 37], we improve the FFN and propose an efficient partial convolution-based feed-forward network (PCFN) to further refine the representative features derived from the SMFA.

Figure 2(c) shows that the PCFN uses a 1×1 convolution with the GELU activation function to perform cross-channel interaction on the expanded hidden space. Then, it splits the hidden features into two chunks of $\{F_\rho^1, F_\rho^2\}$ and employs a 3×3 convolution followed by a GELU activation to process F_ρ^1 to encode local contextual information. The processed F_ρ^1 and F_ρ^2 are then concatenated and fed into a 1×1 convolution for further feature mixing and reducing the hidden channel back to the original input dimension. This process is defined as:

$$\begin{aligned} \{F_\rho^1, F_\rho^2\} &= \mathcal{S}(\phi(Conv_{1\times1}(||F_\rho||_2))), \\ \hat{F}_\rho &= Conv_{1\times1}(\mathcal{C}([\phi(Conv_{3\times3}(F_\rho^1)), F_\rho^2])), \end{aligned} \tag{7}$$

where $F_\rho^1 \in \mathbb{R}^{H\times W\times C/2}$ and $F_\rho^2 \in \mathbb{R}^{H\times W\times 3C/2}$, $\mathcal{C}(\cdot)$ represents a concatenation operation, and $Conv_{3\times3}(\cdot)$ denotes a 3×3 convolutional layer.

3.4 Feature modulation block

As mentioned previously, we formulate the proposed SMFA and PCFN into a feature modulation block (FMB) to produce deep representative features. To stabilize the model training and encourage more information flow, the residual connection is employed. The FMB can be written as:

$$\begin{aligned} F_\rho &= SMFA(F_{in}) + F_{in}, \\ \hat{F}_\rho &= PCFN(F_\rho) + F_\rho, \end{aligned} \tag{8}$$

where F_{in}, F_ρ and $\hat{F}_\rho$ remain consistent with the previous definition.

4 Experimental Results

4.1 Dataset and implementation

Datasets. The high-quality DIV2K [38] and DIV2K + Flickr2K (DF2K) [38] datasets are commonly used for training SR models, and previous methods [14,17,25,26,36] chose one of them as the training dataset. To make a fair comparison with recent state-of-the-art methods, we train our proposed method on the DIV2K and DF2K datasets, respectively. In addition, we evaluate our method on the commonly used test datasets, including Set5 [3], Set14 [46], B100 [2], Urban100 [16] and Manga109 [33]. We transform the image into YCbCr color space and calculate the peak signal-to-noise ratio (PSNR) and structure similarity index (SSIM) on the Y channel of the image to evaluate the quality of the recovered image.

Implementation details. During the training, we randomly crop 64 patches of size 64×64 with random horizontal flips and rotations from LR images as the basic training inputs. The proposed model is trained with the same loss function as SAFMN [36] and optimized by Adam [19] with $\beta_1 = 0.9$ and $\beta_2 = 0.99$. We set the initial learning rate to 1×10^{-3} and the minimum one to 1×10^{-5}, which is updated by the Cosine Annealing scheme [30]. The number of iterations for all experiments is set to 1,000,000. All experiments are conducted with the PyTorch framework on an NVIDIA GeForce RTX 3090 GPU. We train two versions of SMFANet with different complexity. The standard SMFANet consists of 8 FMBs with 36 channels while the large version, SMFANet+, has 12 FMBs with 48 channels. Due to the page limit, we include more experimental results in the supplemental material.

4.2 Comparisons with state-of-the-art methods

Quantitative comparison. To fully evaluate the performance of our method, we first compare our SMFANet with state-of-the-art CNN-based lightweight SR methods, including FSRCNN [10], CARN [1], EDSR-baseline [26], IMDN [17], LAPAR-A [22], SMSR [42], ShuffleMixer [37], SAFMN [36], and further compare the SMFANet+ with recent ViT-based lightweight algorithms, including ESRT [31], SwinIR-light [25], ELAN-light [49], HPINet-S [27], NGswin [7], SPIN [47] and SRFormer-light [53]. The quantitative comparisons on benchmark datasets for the upscaling factors of $\times 2$, $\times 3$, and $\times 4$ are reported in Table 1 and Table 2, where SMFANet (DIV2K) denotes training using the DIV2K dataset, and SMFANet (DF2K) uses DF2K as the training dataset. In addition to PSNR and SSIM metrics, we also list the number of parameters (#Params) and FLOPs (#FLOPs). For a fair comparison, we calculate the model complexity of all evaluated methods with the fvcore library (*i.e.*, fvcore.nn.flop_count_str) under a setting of super resolving an LR image to 1280×720 pixels.

Benefiting from the ability to model non-local information of the SMFA module, SMFANet can effectively explore more information compared to previous

Table 1: Comparison with CNN-based lightweight SR methods on public benchmark datasets. All PSNR/SSIM results are calculated on the **Y**-channel. #FLOPs is measured corresponding to an HR image of the size 1280×720 pixels. The best and second-best performances are highlighted in Red and Blue colors.

Scale	Methods	#Params (K)	#FLOPs (G)	Set5	Set14	B100	Urban100	Manga109
×2	FSRCNN [10]	12	6	37.00/0.9558	32.63/0.9088	31.53/0.8920	29.88/0.9020	36.67/0.9694
	CARN [1]	1592	223	37.76/0.9590	33.52/0.9166	32.09/0.8978	31.92/0.9256	38.36/0.9765
	EDSR-baseline [26]	1370	316	37.99/0.9604	33.57/0.9175	32.16/0.8994	31.98/0.9272	38.54/0.9769
	IMDN [17]	694	161	38.00/0.9605	33.63/0.9177	32.19/0.8996	32.17/0.9283	38.88/0.9774
	LAPAR-A [22]	548	171	38.01/0.9605	33.62/0.9183	32.19/0.8999	32.10/0.9283	38.67/0.9772
	SMSR [42]	985	132	38.00/0.9601	33.64/0.9179	32.17/0.8990	32.19/0.9284	38.76/0.9771
	ShuffleMixer [37]	394	91	38.01/0.9606	33.63/0.9180	32.17/0.8995	31.89/0.9257	38.83/0.9774
	SAFMN [36]	228	52	38.00/0.9605	33.54/0.9177	32.16/0.8995	31.84/0.9256	38.71/0.9771
	SMFANet (DIV2K)	186	41	38.04/0.9606	33.65/0.9186	32.19/0.8999	32.22/0.9283	38.98/0.9776
	SMFANet (DF2K)	186	41	38.08/0.9607	33.65/0.9185	32.22/0.9002	32.20/0.9282	39.11/0.9779
×3	FSRCNN [10]	12	5	33.16/0.9140	29.43/0.8242	28.53/0.7910	26.43/0.8080	30.98/0.9212
	CARN [1]	1592	119	34.29/0.9255	30.29/0.8407	29.06/0.8034	28.06/0.8493	33.50/0.9440
	EDSR-baseline [26]	1555	160	34.37/0.9270	30.28/0.8417	29.09/0.8052	28.15/0.8527	33.45/0.9439
	IMDN [17]	703	72	34.36/0.9270	30.32/0.8417	29.09/0.8046	28.17/0.8519	33.61/0.9445
	LAPAR-A [22]	594	114	34.36/0.9267	30.34/0.8412	29.11/0.8054	28.15/0.8523	33.51/0.9441
	SMSR [42]	993	68	34.40/0.9270	30.33/0.8412	29.10/0.8050	28.25/0.8536	33.68/0.9445
	ShuffleMixer [37]	415	43	34.40/0.9272	30.37/0.8423	29.12/0.8051	28.08/0.8498	33.69/0.9448
	SAFMN [36]	233	23	34.34/0.9267	30.33/0.8418	29.08/0.8048	27.95/0.8474	33.52/0.9437
	SMFANet (DIV2K)	191	19	34.46/0.9275	30.39/0.8432	29.13/0.8059	28.25/0.8525	33.84/0.9454
	SMFANet (DF2K)	191	19	34.42/0.9274	30.41/0.8430	29.16/0.8065	28.22/0.8523	33.96/0.9460
×4	FSRCNN [10]	12	5	30.71/0.8657	27.59/0.7535	26.98/0.7150	24.62/0.7280	27.90/0.8517
	CARN [1]	1592	91	32.13/0.8937	28.60/0.7806	27.58/0.7349	26.07/0.7837	30.47/0.9084
	EDSR-baseline [26]	1518	114	32.09/0.8938	28.58/0.7813	27.57/0.7357	26.04/0.7849	30.35/0.9067
	IMDN [17]	715	41	32.21/0.8948	28.58/0.7811	27.56/0.7353	26.04/0.7838	30.45/0.9075
	LAPAR-A [22]	659	94	32.15/0.8944	28.61/0.7818	27.61/0.7366	26.14/0.7871	30.42/0.9074
	SMSR [42]	1006	42	32.12/0.8932	28.55/0.7808	27.55/0.7351	26.11/0.7868	30.54/0.9085
	ShuffleMixer [37]	411	28	32.21/0.8953	28.66/0.7827	27.61/0.7366	26.08/0.7835	30.65/0.9093
	SAFMN [36]	240	14	32.18/0.8948	28.60/0.7813	27.58/0.7359	25.97/0.7809	30.43/0.9063
	SMFANet (DIV2K)	197	11	32.25/0.8956	28.67/0.7825	27.61/0.7371	26.19/0.7861	30.72/0.9097
	SMFANet (DF2K)	197	11	32.25/0.8956	28.71/0.7833	27.64/0.7377	26.18/0.7862	30.82/0.9104

CNN-based methods. Table 1 shows that our SMFANet achieves better performance on almost all benchmark datasets. For instance, on the ×4 Manga109 dataset, SMFANet (DIV2K) outperforms IMDN [17] by 0.27dB with only 27% FLOPs. When training SMFANet on the larger DF2K dataset, the performance gain is further improved by 0.1dB.

Table 2 shows that the proposed SMFANet+ achieves competitive performance with lower model complexity. Note that ViT-based methods are based on the self-attention mechanism to model long-range dependency, which consumes significant computational costs. In contrast, our SMFA module, with only a few low-consumption operators, efficiently explores non-local information for high-quality reconstruction. On the five benchmark datasets, SMFANet+ (DIV2K) shows similar performance to SwinIR-light [25] with only 43% FLOPs, and SMFANet+ (DF2K) improves the average PSNR gain of the ×4 SR to 0.14dB. These comparisons suggest that our proposed SMFANet can exploit more information than previous CNN-based and ViT-based approaches.

Qualitative comparisons. We compare the visual results of our proposed SMFANet and SMFANet+ with CNN-based methods and ViT-based methods on the ×4 Urban100 dataset, respectively. The evaluated CNN-based methods [17, 22, 26, 36, 37] are listed at the top of Figure 3, which suffer from blurring artifacts and distorted lines. In contrast, our method can restore more accurate

Table 2: Comparison with ViT-based lightweight SR methods on public benchmark datasets. All measured metrics are calculated in the same way as in Table 1. The best and second-best performances are highlighted in Red and Blue colors.

Scale	Methods	#Params (K)	#FLOPs (G)	Set5	Set14	B100	Urban100	Manga109
×2	ESRT [31]	678	1116	38.03/0.9600	33.75/0.9184	32.25/0.9001	32.58/0.9318	39.12/0.9774
	SwinIR-light [25]	910	244	38.14/0.9611	33.86/0.9206	32.31/0.9012	32.76/0.9340	39.12/0.9783
	ELAN-light [49]	621	203	38.17/0.9611	33.94/0.9207	32.30/0.9012	32.76/0.9340	39.11/0.9782
	NGswin [7]	998	146	38.05/0.9610	33.79/0.9199	32.27/0.9008	32.53/0.9324	38.97/0.9777
	SPIN [47]	497	114	38.20/0.9615	33.90/0.9215	32.31/0.9015	32.79/0.9340	39.18/0.9784
	SMFANet+ (DIV2K)	480	108	38.18/0.9611	33.82/0.9202	32.28/0.9011	32.64/0.9323	39.25/0.9777
	SMFANet+ (DF2K)	480	108	38.19/0.9611	33.92/0.9207	32.32/0.9015	32.70/0.9331	39.46/0.9787
×3	ESRT [31]	770	835	34.42/0.9268	30.43/0.8433	29.15/0.8063	28.46/0.8574	33.95/0.9455
	SwinIR-light [25]	918	114	34.62/0.9289	30.54/ 0.8463	29.20/0.8082	28.66/0.8624	33.98/0.9478
	ELAN-light [49]	629	90	34.61/0.9288	30.55/ 0.8463	29.21/0.8081	28.69/0.8624	34.00/0.9478
	NGswin [7]	1007	66	34.52/0.9282	30.53/0.8456	29.19/0.8078	28.52/0.8603	33.89/0.9470
	SPIN [47]	569	58	34.65/0.9293	30.57/0.8464	29.23/0.8089	28.71/0.8627	34.24/0.9489
	SMFANet+ (DIV2K)	487	48	34.63/0.9285	30.52/0.8456	29.23/0.8084	28.59/0.8594	34.17/0.9478
	SMFANet+ (DF2K)	487	48	34.66/0.9292	30.57/0.8461	29.25/0.8090	28.67/0.8611	34.45/0.9490
×4	ESRT [31]	752	298	32.19/0.8947	28.69/0.7833	27.69/0.7379	26.39/0.7962	30.75/0.9100
	SwinIR-light [25]	930	65	32.44/0.8976	28.77/0.7858	27.69/0.7406	26.47/0.7980	30.92/0.9151
	ELAN-light [49]	640	54	32.43/0.8975	28.78/0.7858	27.69/0.7406	26.54/0.7982	30.92/0.9150
	HPINet-S [27]	463	88	32.47/0.8987	28.80/0.7872	27.69/0.7416	26.59/0.8016	30.92/0.9143
	NGswin [7]	1019	40	32.33/0.8963	28.78/0.7859	27.66/0.7396	26.45/0.7963	30.80/0.9128
	SPIN [47]	555	42	32.48/0.8983	28.80/0.7862	27.70/0.7415	26.55/0.7998	30.98/0.9156
	SRFormer-light [53]	873	63	32.51/0.8988	28.82/0.7872	27.73/0.7422	26.67/0.8032	31.17/0.9165
	SMFANet+ (DIV2K)	496	28	32.43/0.8979	28.77/0.7849	27.70/0.7400	26.45/0.7943	31.06/0.9138
	SMFANet+ (DF2K)	496	28	32.51/0.8985	28.87/0.7872	27.74/0.7412	26.56/0.7976	31.29/0.9163

Table 3: Memory and running time comparisons on ×4 SR. #GPU Mem. denotes the maximum GPU memory consumption during the inference phase, derived with the Pytorch torch.cuda.max_memory_allocated() function. #Avg.Time is the average running time on 500 LR images of 320 × 180 pixels. The best and second-best performances are highlighted in Red and Blue colors.

Type	Methods	Quality Metrics		Type	Methods	Quality Metrics	
		#GPU Mem. [M]	#Avg. Time [ms]			#GPU Mem. [M]	#Avg. Time [ms]
CNN	CARN [1]	702.07	13.54	ViT	SwinIR-light [25]	342.44	177.06
	EDSR-baseline [26]	507.13	15.21		ELAN-light [49]	241.34	42.75
	IMDN [17]	204.27	9.64		HPINet-S [27]	445.92	140.45
	LAPAR-A [22]	1811.46	24.40		NGswin [7]	372.85	108.25
	ShuffleMixer [37]	474.79	19.21		SPIN [47]	441.52	798.44
	SAFMN [36]	65.26	8.44		SRFormer-light [53]	320.95	145.99
	SMFANet	93.20	9.26		SMFANet+	247.51	17.39

parallel patterns with sharper boundaries. The bottom of Figure 3 shows that ViT-based methods [7,25,27,31,49] do not restore the structures well, *e.g.*, strips, while our SMFANet+ recovers an image with better structures.

Memory and running time comparisons. To fully demonstrate the efficiency of our proposed SMFANet, we further compare our method with the CNN-based [1, 17, 22, 26, 36, 37] and ViT-based [7, 25, 27, 47, 49, 53] methods in terms of GPU memory consumption and inference time on ×4 SR. Table 3 shows the comparisons of GPU memory and inference time. Our SMFANet has faster inference than the listed CNN methods except SAFMN, and the larger SMFANet+ presents a significant efficiency advantage over ViT-based methods. The running time of SMFANet+ is 17.39ms, which is ×**10** times faster than SwinIR-light [25] and ×**6** times faster than NGswin [7]. Moreover, when compared to ELAN-light [49], which uses a shared group-wise multi-scale self-

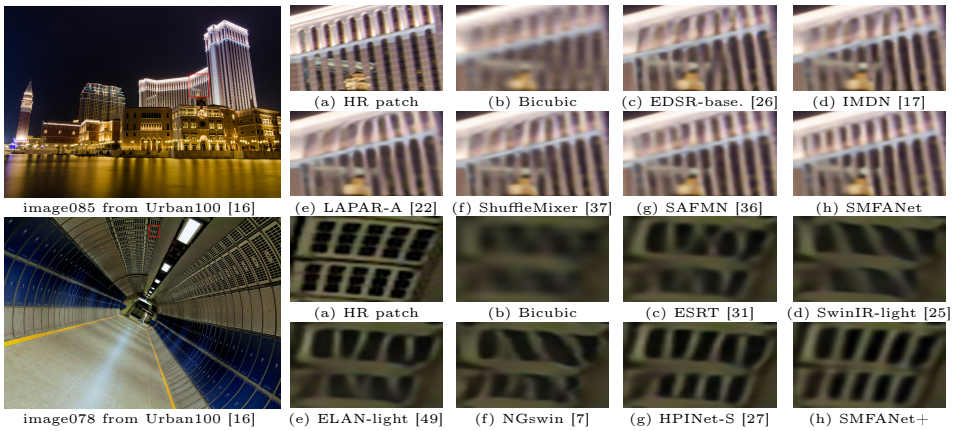


Fig. 3: Visual comparisons for $\times 4$ SR on the Urban100 dataset. The comparisons between SMFANet and CNN-based methods are shown at the top of the Figure, while the comparisons between SMFANet+ and ViT-based methods are exhibited at the bottom. Our proposed method recovers the image with more accurate structures.

attention to accelerate model efficiency, the SMFANet+ still has a much faster runtime (17.39ms *vs.* 42.75ms). These quantitative and qualitative comparison results show that our method obtains even better results against advanced SR models but with significantly less complexity.

LAM comparisons. The local attribution map (LAM) [13] indicates the significant correlation between the red pixels and the rectangular position patch during the restoration process. We compare the LAM [13] results between our method and the evaluated CNN-based [17, 36, 37] and ViT-based [25, 27, 49] methods in Figure 4, and label the corresponding diffusion index (DI) value below each subgraph, where a larger DI value indicates a wider range of pixels involved. These results demonstrate that the proposed SMFANet can explore more non-local information for accurate image super-resolution.

5 Analysis and Discussions

In this section, we conduct extensive ablation studies to analyze and evaluate the effect of each component in the proposed SMFANet. We implement all ablation experiments based on the $\times 4$ SMFANet model and train them with the DIV2K [38] dataset for fair comparisons. The quantitative ablation results in Table 4 are measured on the Urban100 [16] and Manga109 [33] datasets.

Effectiveness of the SMFA. The proposed SMFA module utilizes a parallel structure that absorbs local and non-local feature extraction to enhance reconstruction accuracy. To demonstrate its effectiveness, we first disable the SMFA module to compare it with the baseline SMFANet. Table 4 shows that the PSNR

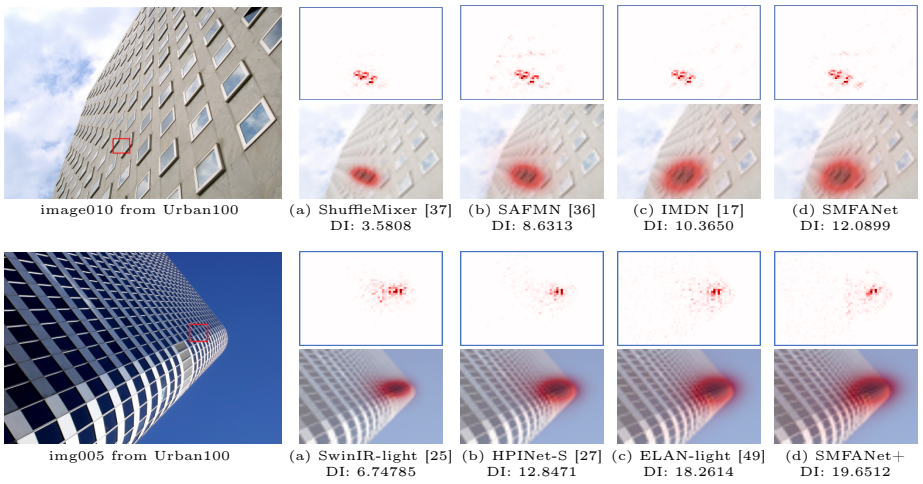
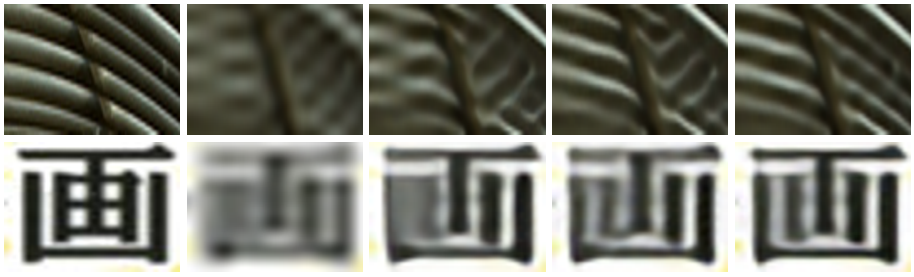


Fig. 4: Comparison of local attribution maps (LAMs) [13] and diffusion indices (DIs) [13]. Our proposed SMFANet family exploit more feature information and reconstructs a more accurate image structure.

values drop by 0.65dB and 0.96dB on the Urban100 and Manga109 datasets. These results show the importance of the SMFA.

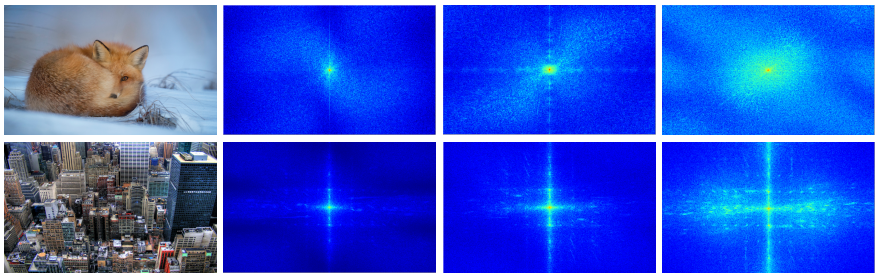
In addition, as the proposed SMFA module mainly contains an EASA branch to explore non-local information and a LDE branch to capture local features, we conduct ablation studies with respect to these components to demonstrate their effectiveness on image SR. Without the EASA branch, significant performance drops of 0.25dB and 0.33dB are observed on the Urban100 and Manga109 datasets. Figure 5 (c) shows the corresponding visual results with blurred structure. Moreover, replacing EASA with the window-based SA [25] yields almost the same performance, but the corresponding runtime and GPU consumption are three times higher. These results indicate that the EASA branch can efficiently approximate SA to explore non-local information for accurate structural reconstruction. As for the LDE branch, removing it, the model only achieves the performance of 25.92dB and 30.22dB on the Urban100 and Manga109. Figure 5 (d) shows the reconstructed images with distorted lines, suggesting that the LDE branch contributes to recovering image details. Figure 6 shows the power spectral density maps, which intuitively illustrate the complementary roles of the EASA and LDE. The PSD of X_l derived from the EASA shows that the energy density is more concentrated in the center region, whereas the PSD of Y_d obtained from the LDE scatters in the surrounding area compared to F_{in} and X_l.

To further validate the effectiveness of this complementary two-branch design, we replace SMFA with either dual LDE branches or dual EASA branches. With these two alternatives, the PSNR values on the Urban100 dataset drop significantly by 0.17dB and 0.25dB, respectively. We also investigated experiments



(a) HR patch (b) LR patch (c) w/o EASA (d) w/o LDE (e) SMFANet

Fig. 5: Effectiveness of the SMFA for image super-resolution on the ×4 Urban100 and Manga109 datasets.



HR image Input feature F_{in} X_l from EASA Y_d from LDE

Fig. 6: The power spectral density (PSD) visualizations of feature F_{in}, X_l, Y_d. We perform a periodic shift of the spectrum map such that the low-frequency component is moved to the center. The EASA activates more low-frequency components for feature X_l, and the LDE enhances high-frequency representations for feature Y_d.

that sequentially process the EASA and LDE, whose corresponding reconstruction performances are clearly decreased as well. These reduced results prove the effectiveness of such a parallel design.

Effectiveness of the variance modulation. The variance modulation is used in the EASA branch to facilitate better non-local information exploration. To demonstrate its effectiveness, we remove this operation from the EASA branch. Table 5 presents that the PSNR values drop by 0.06dB and 0.08dB on the Urban100 and Manga109 datasets. For investigating the effect of $\sigma^2(X)$, we further replace $\sigma^2(X)$ with the variance of X_s and a learnable parameter $\alpha \in \mathbb{R}^{1\times1\times C}$ for feature modulation, and their PSNR values on the Urban100 datasets are reduced to 26.12dB and 26.13dB, respectively. These results suggest that the variance of X_s or the learnable parameter α does not accurately represent the global statistical divergence of the input feature.

In addition, variance modulation with element-wise product incurs a 0.05dB and 0.11dB PSNR reduction on Urban100 and Manga109, respectively. This

Table 4: Ablation experiments for $\times 4$ SMFANet on Urban100 and Manga109 datasets. All measured metrics are calculated in the same way as in Table 1. " A $\rightarrow$ B " is to replace A with B, and "A $\rightarrow$ None" is to remove operator A.

Ablation	Variant	#Params(K)	#FLOPs(G)	#GPU Mem.(M)	#Avg.Time(ms)	Urban100	Manga109
Baseline	SMFANet	**197**	**11**	**93.20**	**9.26**	**26.19/0.7861**	**30.72/0.9097**
SMFA	SMFA $\rightarrow$ None	82	5	56.90	3.12	25.54/0.7646	29.76/0.8974
	EASA $\rightarrow$ None	183	10	77.31	7.28	25.94/0.7783	30.39/0.9061
	EASA $\rightarrow$ Window-based SA	237	16	269.69	37.51	26.20/0.7882	30.74/0.9105
	LDE $\rightarrow$ None	128	7	76.74	6.95	25.92/0.7772	30.22/0.9032
	SMFA $\rightarrow$ Dual LDE branches	263	14	85.72	9.63	26.02/0.7810	30.52/0.9076
	SMFA $\rightarrow$ Dual EASA branches	143	8	88.45	9.28	25.94/0.7783	30.29/0.9076
	SMFA $\rightarrow$ "EASA then LDE"	187	11	77.16	8.90	26.04/0.7824	30.57/0.9077
	SMFA $\rightarrow$ "LDE then EASA"	187	11	72.98	8.95	26.11/0.7841	30.64/0.9083
Variance Modulation	Variance $\sigma^2(X) \rightarrow$ None	197	11	93.20	8.87	26.13/0.7852	30.64/0.9091
	Variance $\sigma^2(X) \rightarrow \sigma^2(X_s)$	197	11	93.20	9.31	26.12/0.7839	30.60/0.9086
	Variance $\sigma^2(X) \rightarrow$ Learnable parameter α	197	11	93.20	9.18	26.13/0.7853	30.65/0.9092
	Addition $\rightarrow$ Element-wise Product	197	11	93.20	9.10	26.14/0.7841	30.61/0.9083
PCFN	PCFN $\rightarrow$ None	132	7	92.94	6.64	25.89/0.7770	30.38/0.9051
	PCFN $\rightarrow$ FFN	174	9	93.11	8.71	26.06/0.7825	30.56/0.9079
	PCFN $\rightarrow$ ConvFFN	213	12	93.13	11.03	26.19/0.7861	30.71/0.9097
	PCFN $\rightarrow$ GDFN	204	12	193.48	10.83	26.22/0.7872	30.75/0.9101

performance drop is caused by the element-wise product operation suppressing the feature channel with low variance, resulting in a decreased information flow.

Effectiveness of the PCFN. The PCFN is used to facilitate the fusion of the representative features extracted by the SMFA module. To demonstrate the effectiveness of this module, we performed ablation experiments with removal of PCFN and replacement of FFN [40], respectively. Table 5 shows that the exclusion of PCFN and the substitution of FFN each leads to a respective decrease in PSNR values of 0.34dB and 0.16dB on the Manga109 [33] dataset, compared to the baseline.

Recent FFN variants [36, 43, 45, 53] introduce convolution to boost local encoding capability for better reconstruction performance. However, the expanded hidden features contain redundant information, and processing them with convolution directly increases the computational cost. For example, we use GDFN [45] for channel mixing, resulting in a $\times 2$ increase in GPU consumption over PCFN, with only a slight improvement in reconstruction performance.

Thus, our PCFN introduces a partial convolution that processes only $1/4$ of the hidden channels, which reduces redundant computation and memory usage while efficiently encoding local information. Table 4 shows that PCFN achieves comparable SR performance with less computational overhead compared to ConvFFN [53] and GDFN [45].

6 Conclusion

We have proposed a simple yet effective SMFANet method to solve image super-resolution efficiently. As the key component of SMFANet, our SMFA module contains the EASA for non-local information exploration and the LDE for local detail modeling. Moreover, we develop a PCFN to facilitate the fusion of the non-local and local information extracted by the SMFA module both in the channel and spatial dimensions. Extensive experimental results show that the proposed SMFANet family achieve a favorable trade-off between reconstruction performance and computational efficiency.

Acknowledgements

This work has been partly supported by the National Natural Science Foundation of China (Nos. U22B2049, 62272233, 62332010), and the Fundamental Research Funds for the Central Universities (No. 30922010910).

References

1. Ahn, N., Kang, B., Sohn, K.A.: Fast, accurate, and lightweight super-resolution with cascading residual network. In: ECCV (2018)
2. Arbeláez, P., Maire, M., Fowlkes, C.C., Malik, J.: Contour detection and hierarchical image segmentation. PAMI **33**(5), 898–916 (2011)
3. Bevilacqua, M., Roumy, A., Guillemot, C., line Alberi Morel, M.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In: BMVC (2012)
4. Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., Gao, W.: Pre-trained image processing transformer. In: CVPR (2021)
5. Chen, J., hong Kao, S., He, H., Zhuo, W., Wen, S., Lee, C.H., Chan, S.H.G.: Run, don't walk: Chasing higher flops for faster neural networks. In: CVPR (2023)
6. Chen, X., Wang, X., Zhou, J., Dong, C.: Activating more pixels in image super-resolution transformer. In: CVPR (2023)
7. Choi, H., Lee, J., Yang, J.: N-gram in swin transformers for efficient lightweight image super-resolution. In: CVPR (2023)
8. Dai, S., Han, M., Xu, W., Wu, Y., Gong, Y.: Soft edge smoothness prior for alpha channel super resolution. In: CVPR (2007)
9. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. PAMI **38**(2), 295–307 (2016)
10. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: ECCV (2016)
11. Dong, J., Pan, J., Yang, Z., Tang, J.: Multi-scale residual low-pass filter network for image deblurring. In: ICCV (2023)
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
13. Gu, J., Dong, C.: Interpreting super-resolution networks with local attribution maps. In: CVPR (2021)
14. Guo, H., Li, J., Dai, T., Ouyang, Z., Ren, X., Xia, S.T.: Mambair: A simple baseline for image restoration with state-space model. arXiv preprint arXiv:2402.15648 (2024)
15. Hendrycks, D., Gimpel, K.: Gaussian error linear units. arXiv preprint arXiv:1606.08415 (2016)
16. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: CVPR (2015)
17. Hui, Z., Gao, X., Yang, Y., Wang, X.: Lightweight image super-resolution with information multi-distillation network. In: ACM MM (2019)
18. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: CVPR (2016)

19. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
20. Li, A., Zhang, L., Liu, Y., Zhu, C.: Feature modulation transformer: Cross-refinement of global representation via high-frequency prior for image super-resolution. In: ICCV (2023)
21. Li, M., Ma, B., Zhang, Y.: Lightweight image super-resolution with pyramid clustering transformer. TCSVT pp. 1–1 (2023)
22. Li, W., Zhou, K., Qi, L., Jiang, N., Lu, J., Jia, J.: LAPAR: Linearly-assembled pixel-adaptive regression network for single image super-resolution and beyond. In: NeurIPS (2020)
23. Li, Y., Fan, Y., Xiang, X., Demandolx, D., Ranjan, R., Timofte, R., Gool, L.V.: Efficient and explicit modelling of image hierarchies for image restoration. In: CVPR (2023)
24. Li, Z., Liu, Y., Chen, X., Cai, H., Gu, J., Qiao, Y., Dong, C.: Blueprint separable residual network for efficient image super-resolution. In: CVPR Workshops (2022)
25. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: SwinIR: Image restoration using swin transformer. In: ICCV Workshops (2021)
26. Lim, B., Son, S., Kim, H., Nah, S., Lee, K.M.: Enhanced deep residual networks for single image super-resolution. In: CVPR Workshops (2017)
27. Liu, J., Chen, C., Tang, J., Wu, G.: From coarse to fine: Hierarchical pixel integration for lightweight image super-resolution. In: AAAI (2023)
28. Liu, J., Tang, J., Wu, G.: Residual feature distillation network for lightweight image super-resolution. In: ECCV Workshops (2020)
29. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin Transformer: Hierarchical vision transformer using shifted windows. In: ICCV (2021)
30. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. In: ICLR (2017)
31. Lu, Z., Li, J., Liu, H., Huang, C., Zhang, L., Zeng, T.: Transformer for single image super-resolution. In: CVPR Workshops (2022)
32. Mao, Y., Zhang, N., Wang, Q., Bai, B., Bai, W., Fang, H., Liu, P., Li, M., Yan, S.: Multi-level dispersion residual network for efficient image super-resolution. In: CVPR Workshops (2023)
33. Matsui, Y., Ito, K., Aramaki, Y., Yamasaki, T., Aizawa, K.: Sketch-based manga retrieval using manga109 dataset. arXiv preprint arXiv:1510.04389 (2015)
34. Park, N., Kim, S.: How do vision transformers work? In: ICLR (2022)
35. Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: CVPR (2016)
36. Sun, L., Dong, J., Tang, J., Pan, J.: Spatially-adaptive feature modulation for efficient image super-resolution. In: ICCV (2023)
37. Sun, L., Pan, J., Tang, J.: ShuffleMixer: An efficient convnet for image super-resolution. In: NeurIPS (2022)
38. Timofte, R., Agustsson, E., Van Gool, L., Yang, M.H., Zhang, L., et al.: NTIRE 2017 challenge on single image super-resolution: Methods and results. In: CVPR Workshops (2017)
39. Timofte, R., DeSmet, V., Van Gool, L.: A+: Adjusted anchored neighborhood regression for fast super-resolution. In: ACCV 2014 (2015)
40. Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, ukasz, L., Polosukhin, Illia: Attention is all you need. In: NeurIPS (2017)
41. Wang, H., Chen, X., Ni, B., Liu, Y., Liu, j.: Omni aggregation networks for lightweight image super-resolution. In: CVPR (2023)

42. Wang, L., Dong, X., Wang, Y., Ying, X., Lin, Z., An, W., Guo, Y.: Exploring sparsity in image super-resolution for efficient inference. In: CVPR (2021)
43. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pvtv2: Improved baselines with pyramid vision transformer. Computational Visual Media **8**(3), 1–10 (2022)
44. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. TIP **19**(11), 2861–2873 (2010)
45. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H.: Restormer: Efficient transformer for high-resolution image restoration. In: CVPR (2022)
46. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Curves and Surfaces (2012)
47. Zhang, A., Ren, W., Liu, Y., Cao, X.: Lightweight image super-resolution with superpixel token interaction. In: ICCV (2023)
48. Zhang, J., Peng, H., Wu, K., Liu, M., Xiao, B., Fu, J., Yuan, L.: Minivit: Compressing vision transformers with weight multiplexing. In: CVPR (2022)
49. Zhang, X., Zeng, H., Guo, S., Zhang, L.: Efficient long-range attention network for image super-resolution. In: ECCV (2022)
50. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: ECCV (2018)
51. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: CVPR (2018)
52. Zhou, L., Cai, H., Gu, J., Li, Z., Liu, Y., Chen, X., Qiao, Y., Dong, C.: Efficient image super-resolution using vast-receptive-field attention. In: ECCV Workshops (2022)
53. Zhou, Y., Li, Z., Guo, C., Bai, S., Cheng, M., Hou, Q.: Srformer: Permuted self-attention for single image super-resolution. In: ICCV (2023)